

# DATABASE SYSTEMS

## Assignment 3: Application Development

공개일	2018년 11월 27일
-----	---------------

문의(조교)	정근성 ( <a href="mailto:aninteger@hanyang.ac.kr">aninteger@hanyang.ac.kr</a> , Discord Geunseong Jung#5285)
--------	---

# 과제 개요

- ‘배달의 한양’ 앱 개발 시나리오
  - 판매자와 구매자, 배달대행자를 연결해주는 앱과 서비스를 개발하는 시나리오
  - 주어진 DB를 기반으로 제시되는 시나리오를 만족하는 앱 개발
- TODO (배점: 기본 점수 150, 추가 점수 50)
  - 사용자 로그인 기능 (10)
  - 판매자(Seller) 화면: 가게(Store) 관리, 메뉴(Menu) 관리, 주문(Order) 관리 (40)
  - 구매자(Customer) 화면: 메뉴로부터 주문 생성, 주문 이력 보기, 배송지 관리 (40)
  - 배달부(Delivery) 화면: 현재 배송 중인 물품 보기 (20)
  - 제출파일: 실행파일, 프로그램 문서 (40)를 하나의 이름\_학번\_DB03.zip파일로 압축하여 제출. (예: 정근성\_2018123456\_DB03.zip)
  - 추가점수: 지도 기능 구현
    - 추가점수는 지난 과제(#1, #2를 총 200점으로 취급)에 더해지며, 이미 두 과제가 만점일 경우 버려짐
- 추가 제출 패널티
  - 기본 제출 기간 내: 100%
  - 추가 제출 기한 내: 50%
  - 추가 제출 기한 이후: 0%

# 시나리오 개괄

- 1명의 Seller는 1개 이상의 Store를 가지고 있으며 각 Store는 1개 이상의 Menu를 판매한다.
- Customer는 Store를 검색하고 그 Store에서 판매하는 여러 개의 Menu를 골라 구매하는 것으로 Order를 생성할 수 있다.
- Order가 생성되면 Seller는 Order를 확인하고 메뉴를 발송한다.
  - Order가 배달부에게 배송이 할당된다.
- Customer는 배달된 물건을 받고(가정하고) 물건을 확인Confirm하는 것으로 Order는 종결된다.

# 로그인

- 로그인은 이메일과 비밀번호를 사용해 진행된다.
  - 이메일과 비밀번호가 일치하면 로그인에 성공한다.
- 판매자, 구매자, 배달부의 로그인 화면은 같은 곳에서 이루어진다.
  - 로그인 화면에서 판매자, 구매자, 배달부를 구분해서는 안 된다.
  - 만약 판매자, 구매자, 배달부의 로그인 정보가 동일한 사용자가 있다면, 로그인 이후 방식은 자유
    - 로그인 이후 어떤 유저 타입을 선택할 건지 정하거나,
    - 가능한 기능을 전부 보여주는 방식 등 본인이 디자인하여 개발

# 판매자

- 판매자 화면:
  - 소유 중인 가게 리스트 표시
  - 개인 정보 변경 기능
    - 비밀번호 변경
    - 이름 변경

# 가게

- 가게 화면:
  - 가게 정보
    - 가게 설명: 구매자들에게 표시되는 가게 설명
    - 가게 태그: 구매자들에게 표시되는 가게 태그 리스트
  - 현재 메뉴
    - 메뉴 추가, 변경, 삭제 기능
  - 현재 주문
    - 주문 확인, 주문 취소 기능
    - 주문 확인 시: 가장 가깝고, 배달 개수Stock가 4개 이하인 배달부를 5명 표시.
      - 배달부를 선택하고 배달 할당.
- 위의 기능이 판매자 화면에 동일하게 나타나도 무방함
  - 단, 어떤 가게의 메뉴, 주문인지 명확하게 구분되어 있어야 함.

# 구매자 (1)

- 구매자 관리 화면
  - 개인 정보 관리 기능: 판매자와 동일
  - 자신의 주소록 목록
    - 주소록 추가, 변경, 제거
- 주문Order 화면
  - 주문 내역
    - 주문한 가게, 메뉴, 결제수단, 주문 시간(타임스탬프)
    - 기본적으로 최근 주문 순으로 정렬
  - 자신의 주문 현황
    - 주문 상태, 주문한 가게, 메뉴, 결제수단, 주문 시간(타임스탬프)
    - 배송 중인 주문은 배달부 정보가 추가되며, 구매자가 배송 완료할 수 있다.
    - 배송 확인된 주문은 주문 내역으로 이동한다.
  - 결제수단 목록
    - 결제수단 추가, 변경, 제거

## 구매자 (2)

- 구매 화면
  - 현재 위치를 기준으로 주변의 영업 중인 가게 리스트 출력 기능
    - 제공되는 DB의 구매자 위도, 경도를 기준으로 리스트만 출력
    - (추가점수) 기기의 현재 위치를 사용하는 코드(Geolocation)을 기준으로 하기
    - (추가점수) 지도를 출력하고 마커를 표시할 경우 추가점수
  - 주소로 가게 검색
  - 태그로 가게 검색
  - 이름으로 가게 검색
- 주문하기
  - 한 가게에서 여러 메뉴와 수량을 골라 장바구니 생성
  - 결제 수단이 있는 주문자의 한해서 결제 활성화
  - 결제하기
    - 결제 수단을 골라 결제
    - 결제 완료 시 주문이 생성되며 결제 시간을 기준으로 주문에 기록
      - 결제 실패는 고려하지 않아도 됨.



# 배달부

- 현재 배송 중인 주문들을 나열한다.
  - 가게, 주문자 정보: 이름, 전화번호, 위치 등 (본인이 판단)
  - 메뉴
  - 주문시간

# Initial DB

- DB \copy command
  - \copy {table\_name} from '{.csv file}' with csv header encoding 'utf8';
- Seller
  - Seller\_id(PK), name, phone, [local, domain] (email for login), passwd
- Store
  - Sid(PK), address, sname, [lat, lng] (store location), phone\_nums (JSON), schedules, seller\_id(FK)
  - Schedules (JSON):
    - Day: 0 for Mon, 1 for Tue, ..., 6 for Sun.
    - Holiday: If True, the store closes that day
    - Open, Closed: open, closing time in a format 'HHMM'

# Initial DB

- Customer
  - Name, phone, [local, domain] (email for login), passwd, [lat, lng] (current location)
  - Payments (JSON)
    - Type: account, card
    - Data:
      - If type is account;  
bid: bank id from *bank*  
acc\_num: account number
      - If type is card;  
card\_num: card number
- Bank
  - Bid(PK), code (bank code), name (bank name)

# Initial DB

- Delivery
  - Did(PK), {same on sellers'}
  - Stock: current # of delivery stock
- Menu
  - Sid(FK) from *store*
  - menu